L19: Entry 5 of 6 File: USPT Feb 13, 1996

DOCUMENT-IDENTIFIER: US 5491531 A
TITLE: Media access controller with a shared class message delivery capability

Brief Summary Text (2):
The present invention relates to digital communications networks, and more particularly, to temporal coordination of when data is exchanged over the network and when a recipients execute commands sent over the network.

Brief Summary Text (4):
Each controller connects to the various sensing devices and operating devices on the machinery which it controls and it automates the operation of that machinery by inputting data from the sensing devices and outputing the appropriate data to the operating devices. This exchange of input and output ("I/O") data between the controller and the devices on the controlled machinery is often accomplished by direct connections between ports on the controller and each separate I/O device. However, when I/O devices are physically spread out over a large area, the wiring costs associated with such separate connections can become excessive.

Brief Summary Text (5):
One way to reduce wiring in such applications is to multiplex I/O data through a serial communications link. Such serial I/O links are commonly used to connect the centrally located controller to separate remote racks, or chassis, that are positioned adjacent to the I/O devices. Such "remote I/O systems" are disclosed, for example, in U.S. Pat. Nos. 4,404,651; 4,413,319; 4,809,217 and 4,750,150, where the controller carries out a continuous high speed scan of the I/O data associated with each remote rack and that data is conveyed through the serial communications link. It can be appreciated that such serial I/O links must promptly and reliably convey the I/O data, since that data is directly controlling operating devices on a machine that is operating in real time.

Brief Summary Text (6):
While a single controller may automate a machine, or a small group of machines, such "islands of automation" must be connected together to truly automate an entire factory. These latter connections are provided by industrial local area networks that enable each controller to send messages to other similar controllers in the factory over serial data links. There are a large number of such "peer-to-peer" communications networks known in the art, some of which are developed by standards committees like IEEE 802.3, IEEE 802.4, IEEE 802.5 and MAP; and others are developed by manufacturers of the controller like those disclosed in U.S. Pat. Nos. 4,319,338; 4,667,323 and 4,747,100. It can be appreciated that the nature of the peer-to-peer data conveyed on local area networks connecting controllers is different than I/O data conveyed on serial I/O links.

Brief Summary Text (7):
A recent trend in industrial control is the increasing use of smaller controllers in so-called "distributed processing." For example, rather than a single large controller linked to a set of remote I/O racks, an alternative architecture is a set of linked small controllers positioned about the factory in place of each remote I/O rack. In a distributed system it may be necessary to coordinate the exchange of data so that a group of these smaller controllers that are associated with a section of the factory will receive data at the same time. It may also be important that data from sensors connected to different controllers be sent over the network at about the same time to provide a "snap shot" of the entire manufacturing process at a given point in time.

Brief Summary Text (8):
A first class of data, such as I/O data, being exchanged is so time critical that it needs to be sent over the network every time that the station at which the data is generated get access to the network. A second class of data, such as production information, is not so time critical and need only be sent once a second, for example. Therefore, it is desirable to provide a mechanism by which the producer of the data is able to determine when the data is released for transmission over the network. Further since a given station may produce data having different degrees of importance, the transmission control mechanism should enable higher priority data to be sent before lower priority data regardless of the order in which the two classes of data were produced.

Brief Summary Text (11):
A digital communication network comprises stations connected to a transmission medium for the exchange of data among the stations. Activity on the network is divided into periodic intervals having known duration. During a first segment of the periodic interval, relatively high priority data is exchanged over the transmission medium. Because of the nature of this data every station that produces high priority data is guaranteed an opportunity to send that class of data. A second segment typically occurs in each periodic interval during which at least some of the stations are afforded an opportunity to transmit lower priority data.

Detailed Description Text (3):
The network 10 is referred to herein as a concurrent time division, multiple access (CTDMA) communications network in that both input/output data used by the processors 13-15 in operating the machinery, as well as control commands, system management and other messages, are concurrently exchanged over the same medium 11 with each node 20 being granted access to transmit messages in a time division multiplexed manner. The CTDMA network 10 utilizes a token bus type protocol in which an "implicit token" is passed from node to node in a prescribed order and the node 20 possessing the implicit token can transmit its messages. However, unlike other token passing networks, a token message is not passed among the nodes, rather each station monitors the network activity and has a register in which it keeps track of the node that it believes currently possesses the token.

Detailed Description Text (4):
Network Communication Protocol

Detailed Description Text (9):
The third and fourth control bits 40 and 42 are designated "tag pad" and "data pad" and indicate the alignment of the tag and data fields 34 and 36 in the Lpacket. As will be described, the tag and data fields have variable lengths and may or may not begin on a sixteen-bit word boundary in the Lpacket 26. The tag pad and data pad control bits 40 and 42 indicate that alignment and are used by a recipient node in separating the Lpacket fields. The next control bit 44 indicates whether the data contained in the Lpacket is in byte size or sixteen-bit word sized increments. The final section 46 comprises three control bits which provide a generation count. In some implementations of the CTDMA protocol a source node may send multiple copies of the same Lpacket to insure that at least one of them will reach the intended recipients. The generation count remains the same for each duplicate Lpacket and enables a recipient node to ignore the duplicates after one has been successfully received. When the generation count changes, the network nodes 20 know that the message contents also has changed.

Detailed Description Text (11):
The final field 36 of the Lpacket 26 contains the data being exchanged, referred to as "link data". The number of bytes in field 36 varies depending upon the amount of data the source node has to send. Since the data may be in byte or word sized increments, receiver circuits in the recipient nodes must be able to handle either format. As will be described, the receiver circuits are designed to handle word size increments. If the link data 36 has an odd number of bytes, as indicated by the data pad control bit 42, the recipient node must add a pad byte to the data field before storing the Lpacket into memory.

Detailed Description Text (14):
During the scheduled phase 52, each node 20 that has been designated as having I/O data is afforded an opportunity to transmit I/O data over the network. As depicted in FIG. 4A, the scheduled phase 52 is broken down into time slots 60 for each of the N nodes 20 on the network 10. As will be described, the scheduled phase can be configured so that not all N nodes will have access in situations where not every node transmits I/O data, such as the programming terminal 18. Each time-slot 60 may be as long as the time required to send a MAC frame 21 of the maximum length allowed by the network protocol (e.g. 510 bytes). Thus the periodic interval 50 must have a defined duration that is at least sufficiently long to permit every node 20 to send the maximum duration MAC frame 21.

Detailed Description Text (18):
During the so called unscheduled phase 54, the network nodes 20 are afforded an opportunity to send any remaining I/O data as well as other message types, such as ones containing control commands and manufacturing production information. Whatever time remains in the periodic interval 50 after the scheduled phase 52 is devoted to the unscheduled phase 54. Therefore each node 20 is not necessarily guaranteed time during the unscheduled phase 54 in which to send messages, however this is acceptable as the unscheduled phase messages by definition are less time critical than the I/O data.

Detailed Description Text (23):
Thus, the present CTDMA network communication protocol allows both input/output data and other message information to be sent over the same network medium in designated portions of the periodic intervals 50. During each periodic interval 50, priority is given to the I/O data which is used to control the machinery operated by the processors 13-15 on the network 10. Once every node 20 on the network has been afforded the opportunity to send its I/O data the remaining portion of the periodic interval is devoted to the transmission of other forms of information during an unscheduled phase. Although each network node 20 is not guaranteed the opportunity to sent a message during the unscheduled phase 54, over the course of several periodic intervals each node will receive that opportunity. If such opportunities do not occur frequently enough, the network operator may have to lengthen the periodic interval.

Detailed Description Text (25):
Referring to FIG. 5A, the CTDMA protocol preferably is implemented on a single, very large scale (VLSI) integrated circuit, designated as the "S-Link Media Access Controller" (SMAC) 100. The SMAC 100 interfaces with the host processor 102 via a conventional bus 101. The bus 101 includes standard address, data and control signals as known in the art for a processor interface. The host processor 102 may be a conventional microprocessor system. The SMAC 100 is preferably configurable to permit adaptation to a variety of host processor bus configurations and protocols. For example, data may selectively be transferred as eight or sixteen bit wide data. Similarly, "handshaking" signals for data transfer may be adapted for known interface protocols used by commercially available microprocessors. A facility is preferably provided by the SMAC 100 to "byte swap" data bytes according to the high/low byte precedence used by the connected host 102. Such interface features are described in more detail in co-pending U.S. Pat. No. 5,153,884 entitled "Intelligent Network Interface Circuit."

Detailed Description Text (30):
The SMAC 100 also provides a SYNC OUT signal 120 which provides a capability for real time synchronization between different nodes 20 connected on the network 10. Because of the unique qualities of the CTDMA protocol, it is possible for each individual node 20 to synchronize activities at the local node to global, real time synchronized events which occur on the network 10. The SYNC OUT signal 120 is configurable to select one of several possible synchronization sources within the SMAC 100. Based on the selected synchronization source, the sync output signal 120 provides a capability to synchronize processes at the local node 20 with respect to an absolute, or "universal" time reference which is available to all nodes on the network. This allows the real time synchronization of events and systems, even though those events and systems may be on different network nodes 20. The local process to be synchronized may be a separate circuit or subsystem, as represented by the dashed lines 121, or alternatively may be a process operating the host processor 102 itself. In the latter case, the SYNC OUT signal 120 may optionally be connected as a discreet input signal into the host

processor 102. Although a single SYNC OUT signal 120 is shown, multiple SYNC OUT signals may be utilized if desired to simultaneously provide other of the selectable "sync event" sources.

Detailed Description Text (40):
The access control circuit 144 coordinates all functions on the SMAC 100, including the implementation of the CTDMA protocol, data transmission, data reception, data exchange with the host processor 102, and the generation of the sync output signal 120. The access control circuit 144 includes a reduced instruction set computer, referred to herein as RISC processor 400 (FIG. 10), in addition to several dedicated logic circuits. The RISC processor 400 is interfaced to a RISC bus 180 which connects the access control circuit 144 to other processing circuits on the SMAC 100. The RISC bus 180 includes address, data and control lines, and provides for communication of command and status information between the RISC processor 400 and the connected circuits. In particular, the RISC processor 400 utilizes the RISC bus 180 to generate command strobes, test individual status bits, and even read "byte" data in parallel, from selected locations within the SMAC 100. While most I/O operations to the access control circuit 144 are made via the RISC bus 180, some discreet inputs and outputs also are employed, and these signals are described where appropriate below.

Detailed Description Text (41):
The access control circuit 144 also produces several signals which are pertinent to operation of the CTDMA protocol, and those signals are grouped together into a bus labeled "sync" bus 176. The sync bus 176 connects to the sync select circuit 145 to provide the source signals from which the sync select circuit 145 can generate the SYNC OUT signal 120. The sync bus 176 also connects to the receive processing circuit 142 and to the host interface 143.

Detailed Description Text (42):
Referring still to FIG. 5B, the transmit processing circuit 141 generates a serial stream of phy symbols for output from the SMAC 100. The serial phy symbol stream and other associated control signals are connected to the modem circuit 140 by the output bus 161. The transmit processing circuit 141 performs the functions of generating the preamble 22, start delimiter 23, data field 25 and end delimiter 28 for an outgoing MAC frame 21. The preamble, start delimiter and end delimiter are fixed in format and are generated internally by the transmit processing circuit 141. The data for transmission in the data field 25 of the MAC frame is provided to the transmit processing circuit 141 either directly from the host interface circuit 143 or from the access control circuit 144 via the RISC bus 180. For "normal" output data (i.e. from the host 102), the data field contents are supplied to the transmit processing circuit 141 directly by the host interface circuit 143. However, another important concept is the capability of the SMAC 100 to automatically respond to certain station management commands, in some cases by actually transmitting an appropriate station management response message. When required, the station management response message is sent without direction from, or disruption of, the host processor 102. The station management response messages are generated automatically by the RISC processor in the access control circuit 144, and the corresponding data for transmission is sent to the transmit processing circuit 141 by way of the RISC bus 180.

Detailed Description Text (44):
The host interface circuit 143 is a "stream" oriented interface for the host processor 102. Each stream is a logical grouping of memory locations organized in a first in, first out (FIFO) manner within the host interface circuit 143. A total of five streams are preferably implemented; three output (transmit) streams and two incoming (receive) streams. The three output streams are designated TX A, TX B and TX C. Data pending transmission is placed in the streams TX A through TX C by the host processor 102 using the host bus 101. A capability is also provided for the host processor 102 to "mark" the output streams as "ready for transmission". A STREAM STATUS bus 185 is provided by the host interface circuit 143 to indicate which of the three output streams have been marked as "ready for transmission". The access control circuit 144 is also appraised of the "ready for transmission" status via the RISC bus 180. When permission to transmit has been acquired by the node 20 according to the CTDMA protocol, the access control circuit 144 issues appropriate instructions over the RISC bus 180 to instruct the transmit processing circuit 141 to commence transmission. At the same time, the RISC bus 180 is also used to designate one of the "ready" streams in the host

interface 143 for actual transmission. The information as to which stream is to be transmitted is latched by the host interface 143, and provided to the transmit processing circuit 141 via a "frame start" bus 186. The host interface circuit 143 then places the first, or top most data word for the selected stream onto a "TX word" bus 187. The TX word bus 187 transfers the current word to the transmit processing circuit 141. The TX word bus 187 is 16 bits wide, such that two bytes at a time are transferred. Once the transmit processing circuit 141 has latched the current word on the TX word bus 187, a TAKE WORD signal 188 is activated. The TAKE WORD signal 188 commands the host interface 143 to advance the currently selected stream to the next data word for transmission.

Detailed Description Text (59):
In addition, the CTDMA protocol allows for the intentional abort of a message frame 21. An intentional abort is signified by receipt of a second start delimiter after the first start delimiter which started the MAC frame. If a second start delimiter sequence is detected, an RX ABORT signal 282 is activated by the delimiter detect circuit 265 and coupled through the input bus 162 to the receive processing circuit 142.

Detailed Description Text (68):
Referring still to FIG. 8, in addition to the input and output signals described above, the delimiter detect state machine 346 also receives as inputs the BYTE PULSE signal 151, the PHY CLOCK signal 150 and a SHUTDOWN signal 369. The PHY CLOCK signal 150 is used as the basic dock for operating the state machine, while the BYTE PULSE signal 151 is used by the state machine in cases where state transitions are made on "byte" boundaries. The SHUTDOWN signal 369 is generated by a decode command instruction circuit 368. The decode command instruction circuit 368 is connected to the RISC bus 180 and includes appropriate logic gates to decode the address and command lines on the RISC bus 180 to detect a predetermined pattern which represents issuance of the "shutdown" command by the RISC processor. In this manner, the RISC processor is able to in effect issue a "software reset" to the delimiter detect state machine 346 when abnormal conditions are detected. The outputs generated by the delimiter detect state machine 346 include the VALID FRAME signal 365 described above, plus the ALIGN BYTE signal 266, the RX READY signal 166 and the BAD FRAME signal 281.

Detailed Description Text (77):
The access control circuit 144 in the SMAC 100 shown in FIG. 5B includes a RISC 400 processor and other specialized support circuits for implementing the CTDMA protocol. As shown in detail in FIG. 10, the access control support circuits 401-407 include an event logic circuit 401, a "divide by ten" pulse generator 402, a MAC frame slot timer 403, a periodic interval timer 404, a guard time monitor 405, a scheduled phase register 406 and a decode circuit 407.

Detailed Description Text (78):
The RISC processor 400 is of conventional construction as is known to those skilled in the art. Tb, e RISC processor 400 includes internal random access memory (RAM) 410 and read only memory (ROM) 411, as well as other known circuits (not shown, e.g. registers, arithmetic/logic unit, program counter, op code decoder, etc.) for executing a program of instructions stored in the ROM 411. The RAM 410 contains a set of storage locations in which is stored the active network protocol parameters, as specified in the Lpacket sent by the moderator during the guard band 56 of each periodic interval 50.

Detailed Description Text (83):
Still referring to FIG. 10, the periodic interval timer 404 is used in conjunction with the RISC processor 400 to indicate the time that elapses during each periodic interval 50. Overall, the time expired in each periodic interval 50 is measured as a sixteen bit count (two bytes), referred to herein as a periodic interval time (PIT) count. The PIT count has a resolution of ten microseconds per bit, which allows a maximum periodic interval of 655.35 milliseconds. The lower eight bit portion of the PIT count (least significant) is maintained within the periodic interval timer 404 itself, while the upper eight bit portion (most significant) is maintained by the RISC processor 400 in the RAM 410. At the beginning of a periodic interval 50, each portion is loaded with a value from RAM 410 that was received from the network moderator in the Lpacket sent during the guard band 56. The lower eight bit count, designated as bus PIT LO BYTE 418, is sent from the periodic interval timer 404 to the guard time monitor 405. Circuits are

also provided within the periodic interval timer 404 to allow the value on the PIT LO BYTE bus 418 to be read directly by the RISC processor 400 via the RISC bus 180. The count on the PIT LO BYTE bus 418 is automatically decremented each ten microseconds until a count of zero is reached. A facility is provided within the periodic interval timer 404 to enable the RISC processor 400 to detect a zero count on the PIT LO BYTE bus 418 via the RISC bus 180. When that occurs, the RISC2 processor 400 decrements the value of the PIT high byte count (in RAM 410) by one. This process repeats until the PIT high order byte is decremented down to zero. At that time, the RISC processor 400 issues appropriate commands over the RISC bus 180 to activate a PIT HI BYTE ZERO signal 419 within the periodic interval timer 404. The PIT HI BYTE ZERO signal thereby indicates that the PIT high byte is now zero, and the only time remaining in the current periodic interval is the time represented by the low byte value on the PIT LO BYTE bus 418. The PIT HI BYTE ZERO signal 419 is also output to the guard time monitor 405.

Detailed Description Text (84):
One additional function of the periodic interval timer 404 is to produce a TONE COMMAND signal 420. The TONE COMMAND signal 420 is activated by the periodic interval timer 404 upon receipt of appropriate commands from the RISC processor 400 via the RISC bus 180. As discussed above with respect to FIG. 3, the "tone" 53 is the internal SMAC reference signal which marks the beginning of a new periodic interval 50, and occurs essentially at the same absolute, or "universal" time in all nodes 20 on the network 10, within a small margin of error due to propagation delays of time reference events. The TONE COMMAND signal 420 is used internally within the periodic interval timer 404 and also connected to the guard time monitor 405, the scheduled phase register 406, and the sync bus 176.

Detailed Description Text (86):
The "guard time" value is configurable, i.e. the guard time register 421 can be read from or written to by the RISC processor 400 via the RISC bus 180. A decode circuit 423 is connected to the RISC bus 180 to detect execution of a "write guard band time" command by RISC processor 400. When the "write guard band time" command is detected, the decode circuit 423 generates a strobe on line 424. The strobe 424 in turn causes the guard time register 421 to latch the data presented on the eight data lines of the RISC bus 180, represented by bus segment 425. Similarly, the RISC processor 400 is able to read back the contents of the guard band time register 421 via a byte select circuit 426. The byte select circuit 426 is connected between the "guard band time" bus 422 and the RISC bus 180, and decodes appropriate control signals on the RISC bus 180 to detect execution of a "read guard band time value" instruction by the RISC processor 400. When the "read guard band time value" instruction is detected, the "guard band time" value on bus 422 is coupled through to the data lines of RISC bus 180.

Detailed Description Text (89):
Flip flop 435 is reset directly by the TONE COMMAND signal 420. The GENERAL GUARD BAND signal 431 is therefore "true" for the entire duration of the guard band 56, i.e. from the time that the PIT count reaches the "guard band time" value until the "tone" 53. The SCREENER GUARD BAND signal 432, however, is reset by an explicit command sent by the RISC processor 400. A decode circuit 437 is connected to the RISC bus 180, and detects the execution of a "reset screener guard band" instruction by the RISC processor 400. When the "reset screener guard band" instruction is detected, a strobe is produced on output line 438. Output line 438 is in turn connected to a reset input on flip flop 436, which when activated causes the flip flop 436 to reset on the next active edge of the PHY CLOCK signal 150. This allows the RISC processor 400 to deactivate the SCREENER GUARD BAND signal 432 at a time other than at the "tone" 53 for operation of the screening function as described below.

Detailed Description Text (90):
The third guard band control signal, namely the MODERATOR GUARD BAND signal 433, is connected as an output of an AND gate 440. One input of AND gate 440 is connected to the GENERAL GUARD BAND signal 431. The other input of AND gate 440 is connected to an output 441 of a flip flop 442. Flip flop 442 is under direct control of the RISC processor 400, and is used to store the "moderator" status for the node 20. If the RISC processor 400 has determined that its own node 20 is the moderator node for the network 10, then flip flop 442 is commanded to the "set" state. In that case, AND gate 440 is enabled, and the MODERATOR GUARD BAND signal 433 follows the GENERAL GUARD BAND signal 431 (i.e. the two are equivalent). Alternatively, if the RISC

processor 400 determines that its own node is not the moderator node, then flip flop 442 is reset, and the MODERATOR GUARD BAND signal 433 is never activated.

Detailed Description Text (91):
Control of flip flop 442 is provided by a decode circuit 443. Decode circuit 443 is connected to the RISC bus 180, and decodes separate "set" and "reset" commands for the "moderator" status of the node 20. Upon detecting execution of either the "set" or "reset" commands, an output strobe is generated on one of two output lines 444 or 445, respectively. The output lines 444 and 445 are in turn connected to respective synchronous "set" and "reset" inputs on flip flop 442.

Detailed Description Text (94):
Now referring to FIGS. 10 and 12, the scheduled phase register circuit 406 of the access control 144 includes a flip flop 448, the output of which is termed a SCHEDULED PHASE signal 450. The SCHEDULED PHASE signal 450, when "true", indicates that the periodic interval 50 is currently in the scheduled phase 52. The flip flop 448 has synchronous set and reset inputs, and is clocked by the PHY CLOCK signal 150. The TONE COMMAND signal 420 is connected to the set input of flip flop 448. The SCHEDULED PHASE signal 450 is therefore automatically set "true" upon activation of the TONE COMMAND signal 420. This forces the scheduled phase to always begin at the "tone" 53, per the CTDMA protocol (see FIG. 3). The flip flop 448 is reset by an appropriate command from the RISC processor 400. Specifically, the scheduled phase register circuit 406 includes a decode circuit 451 connected to the RISC bus 180. The decode circuit 451 decodes appropriate address and control lines on the RISC bus 180 to detect execution of a "reset scheduled phase" output instruction by the RISC processor 400. When the "reset scheduled phase" instruction is detected, an output line 452 connected to the reset input of the flip flop 448 is activated, resetting flip flop 448. Once flip flop 448 is reset (SCHEDULED PHASE signal 450 "false"), the periodic interval 50 is considered to be in the "unscheduled" phase 54 up until the start of the "guard band" as defined above. The SCHEDULED PHASE signal 450 also stays "false" throughout the guard band 56 until again set "true" by the next activation of the TONE COMMAND signal 420.

Detailed Description Text (99):
Referring still to FIG. 13, the TONE COMMAND signal 420 is produced by a decode circuit 525 connected to the RISC bus 180. This circuit 525 decodes execution of a "generate tone" instruction by the RISC processor 400 and emits the TONE COMMAND signal 420 when the "generate tone" instruction is detected. The PIT HI BYTE ZERO signal 419 is maintained by a PIT hi byte zero flip flop 530, that is clocked by the PHY CLOCK signal 150. The flip flop 530 can be set under control of the RISC processor 400. As previously stated, the RISC processor maintains the PIT high order byte internally, i.e. in RAM 410. The RISC processor 400 also takes care of decrementing the PIT high order byte each time the down counter 500 produces a "carry". Each time the RISC processor 400 decrements the PIT high order byte, the result of the decrement operation (i.e. the "zero" or "non-zero" status of the result) is written to the PIT hi byte zero flip flop circuit 530 via a "write PIT hi byte result" instruction on the RISC bus 180. When the TONE COMMAND signal 420 becomes "true", flip flop 530 and the corresponding PIT HI BYTE ZERO signal 419 are forced to a reset condition.

Detailed Description Text (118):
With reference to FIG. 17, the SMAC host interface 143 couples the transmit and receive processing circuits 141 and 142 to the external host processor 102 and handles the exchange of data therebetween.

Detailed Description Text (128):
The output of each transmit FIFO 841-843 is connected via bus 850 to a transmit control circuit 854 that respond to commands from the RISC processor 400. The RISC processor 400 periodically queries the transmit control circuit 854 via the RISC bus 180 to determine whether there is data in any of the transmit FIFO's that is ready to be sent over the network.

CLAIMS:

1. In a digital communication network of the type in which (a) activity on the digital

communication network is divided into periodic time intervals each of which being subdivided into a first phase and a second phase, (b) protocol for operation of the digital communication network guarantees that all active subscribers to the network are granted access to the network during every repetition of the first phase, (c) each first phase transmission is limited to a predetermined maximum time interval, and (d) active subscribers to the network are selectively granted access to the network during the second phase according to a predetermined selection criteria, an improved media access controller comprising:

an interface for connecting to a device that produces data for transmission over the digital communication network;

a first memory connected to said interface for storing a first restricted class of data supplied by the device, the first restricted class of data being restricted to transmission during the first phase only;

a second memory connected to said interface for storing a shared class of data supplied by the device which shared class of data may be transmitted over the digital communication network during the first and second phases;

a transmitter coupling said first and second memories to the network for the transmission of data; and

a control circuit which transfers data from said first and second memories to said transmitter, data being transferred from said first memory when the media access controller has access to the network during one of said first phases, and shared class data being transferred from said second memory at the earliest occurrence of either (1) an opportunity to include the shared class data from said second memory in one of said first phases without exceeding said predetermined maximum transmission time interval after having transmitted said first restricted class of data, or (2) a grant of access to the network is obtained by the media access controller during one of said second phases.

3. A method for operating a media access controller on a digital communication network of the type in which (a) activity on the digital communication network is divided into periodic time intervals each of which being subdivided into a first phase and a second phase, (b) protocol for operation of the digital communication network guarantees that all active subscribers to the network are granted access to the network during every repetition of the first phase, (c) each first phase transmission is limited to a predetermined maximum time interval, and (d) active subscribers to the network are selectively granted access to the network during the second phase according to a predetermined selection criteria, the method comprising the steps of:

(a) accepting and buffering a first restricted class of data from an external interface for transmission only during one of said first phases;

(b) accepting and buffering a shared class of data from the external interface for transmission at the earliest opportunity but without imposing delay on transmission of said first restricted class data;

(c) acquiring access to the digital communications network during the first phase and transmitting the buffered first restricted class data;

(d) if the buffered shared class data can be transmitted after the data of said first restricted class while maintaining access to the network during the first phase without exceeding said predetermined maximum time interval, then transmitting the buffered shared class data during the first phase; and

(e) if network access is acquired during the second phase, and if the buffered shared class data is still awaiting transmission, then transmitting the shared class data.

Previous Doc Next Doc Go to Doc#